

Automatic Security Evaluation and (Related-key) Differential Characteristic Search: Application to SIMON, PRESENT, LBlock, DES(L) and Other Bit-Oriented Block Ciphers^{*}

Siwei Sun^{1,2}, Lei Hu^{1,2}, Peng Wang^{1,2}, Kexin Qiao^{1,2}, Xiaoshuang Ma^{1,2},
and Ling Song^{1,2}

¹State Key Laboratory of Information Security, Institute of Information Engineering,
Chinese Academy of Sciences, Beijing 100093, China

²Data Assurance and Communication Security Research Center,
Chinese Academy of Sciences, Beijing 100093, China

{[sunsiwei](mailto:sunsiwei@iie.ac.cn),[hulei](mailto:hulei@iie.ac.cn),[wpeng](mailto:wpeng@iie.ac.cn),[qiaokexin](mailto:qiaokexin@iie.ac.cn),[maxiaoshuang](mailto:maxiaoshuang@iie.ac.cn),[songling](mailto:songling@iie.ac.cn)}@iie.ac.cn

Abstract. We propose two systematic methods to describe the differential property of an S-box with linear inequalities based on logical condition modelling and computational geometry respectively. In one method, inequalities are generated according to some conditional differential properties of the S-box; in the other method, inequalities are extracted from the H-representation of the convex hull of all possible differential patterns of the S-box. For the second method, we develop a greedy algorithm for selecting a given number of inequalities from the convex hull. Using these inequalities combined with Mixed-integer Linear Programming (MILP) technique, we propose an automatic method for evaluating the security of bit-oriented block ciphers against the (related-key) differential attack with several techniques for obtaining tighter security bounds, and a new tool for finding (related-key) differential characteristics automatically for bit-oriented block ciphers.

Keywords: Automatic cryptanalysis, Related-key differential attack, Mixed-integer Linear Programming, Convex hull.

1 Introduction

Differential cryptanalysis [7] is one of the most well-known attacks on modern block ciphers, based on which many cryptanalytic techniques have been developed, such as truncated differential attack [34], impossible differential attack [9], and boomerang attack [51]. Providing a security evaluation with respect to the differential attack has become a basic requirement for a newly designed practical block cipher to be accepted by the cryptographic community.

* An extended version of this paper containing more applications and the source code is available at <http://eprint.iacr.org/2013/676>.

Contrary to the single-key model, where methodologies for constructing block ciphers provably resistant to differential attacks are readily available, the understanding of the security of block ciphers with regard to related-key differential attacks is relatively limited. This limited understanding of the security concerning related-key differential attacks has been greatly improved in recent years for AES-like byte- or word-oriented SPN block ciphers. Along this line of research, two representative papers [10,25] were published in Eurocrypt 2010 and Crypto 2013. In the former paper [10], an efficient search tool for finding differential characteristics both in the state and in the key was presented, and the best differential characteristics were obtained for some byte-oriented block ciphers such as AES, byte-Camellia, and Khazad. In the latter paper [25], Pierre-Alain Fouque *et al.* showed that the full-round AES-128 can not be proven secure against differential attacks in the related-key model unless the exact coefficients of the MDS matrix and the S-Box differential properties are taken into account. Moreover, a variant of Dijkstra's shortest path algorithm for finding the most efficient related-key attacks on SPN ciphers was developed in [25]. In [27], Ivica Nikolic presented a tweak for the key schedule of AES and the new cipher called xAES is resistant against the related-key differential attacks found in AES.

For bit-oriented block ciphers such as PRESENT-80 and DES, Sareh Emami *et al.* proved that no related-key differential characteristic exists with probability higher than 2^{-64} for the *full-round* PRESENT-80, and therefore argue that PRESENT-80 is secure against basic related-key differential attacks [22]. In [48], Sun *et al.* obtained tighter security bounds for PRESENT-80 with respect to the related-key differential attacks using the Mixed-integer Linear Programming (MILP) technique. Alex Biryukov and Ivica Nikolić proposed two methods [11] based on Matsui's tool [42] for finding related-key differential characteristics for DES-like ciphers. For their methods, they stated that "... our approaches can be used as well to search for high probability related-key differential characteristics in any bit-oriented ciphers *with linear key schedule*."

Sareh Emami *et al.* [22] and Sun *et al.*'s method [48] can not be used to search for actual (related-key) differential characteristics, and Alex Biryukov *et al.*'s method [11] is only applicable to ciphers with linear key schedule.

In this paper, we provide a method based on MILP which can not only evaluate the security (obtain security bound) of a block cipher with respect to the (related-key) differential attacks, but is also able to search for actual (related-key) differential characteristics even if the key schedule algorithm of the block cipher is nonlinear.

The problem of MILP is a class of optimization problems derived from Linear Programming in which the aim is to optimize an objective function under certain constraints. Despite its intimate relationship with discrete optimization problems, such as the set covering problem, 0-1 knapsack problem, and traveling salesman problem, it is only in recent years that MILP has been explicitly applied in cryptographic research [1,17,18,36,46,52,57].

In this paper, we are mainly concerned with the application of MILP method in the (related-key) differential cryptanalysis. A practical approach to evaluate

the security of a cipher against differential attack is to determine the lower bound of the number of active S-boxes throughout the cipher. This strategy has been employed in many designs [4,8,15,16,19]. MILP was applied in automatically determining the lower bounds of the numbers of active S-boxes for some word-oriented symmetric-key ciphers, and therefore used to prove their security against differential cryptanalysis [14,44,54]. Laura Winnen [53] and Sun *et al.* [48] extended this method by making it applicable to ciphers involving bit-oriented operations. We notice that such MILP tools [14,44,48,54] for counting the minimum number of active S-boxes are also applied or mentioned in the design and analysis of some authenticated encryption schemes [8,20,21,29,30,31,55,58].

Our Contributions. We find that the constraints presented in [48] are too coarse to accurately describe the differential properties of a specific cipher, since there are a large number of invalid differential patterns of the cipher satisfying all these constraints, which yields a feasible region of the MILP problem much larger than the set of all valid differential characteristics.

In this paper, we propose two methods to tighten the feasible region by cutting off some impossible differential patterns of a specific S-box with linear inequalities: one method is based on logical condition modeling, and the other is a more general approach based on convex hull computation — a fundamental algorithmic problem in computational geometry.

However, the second approach produces too many inequalities so that adding all of them to an MILP problem will make the solving process impractical. Therefore, we develop a greedy algorithm for selecting a given number of linear inequalities from the convex hull.

By adding all or a part of the constraints generated by these methods, we provide MILP based methods for evaluating the security of a block cipher with respect to the (related-key) differential attack, and searching for actual (related-key) differential characteristics. Using these methods, we obtain the following results.

1. The probability of the best related-key differential characteristic of the 24-round PRESENT-80 is upper bounded by 2^{-64} , which is the tightest security bound obtained so far for PRESENT-80.
2. The probability of the best related-key differential characteristic for the full-round LBlock is at most 2^{-60} .
3. We obtain a single-key *differential characteristic* and a single-key *differential* for the 15-round SIMON48 (a lightweight block cipher designed by the U.S. National Security Agency) with probability 2^{-46} and $2^{-41.96}$ respectively, which are the best results published so far for SIMON48.
4. We obtain a 14-round related-key differential characteristic of LBlock with probability 2^{-49} *in no more than 4 hours on a PC*. Note that the probabilities of the best previously published related-key characteristics covering the 13- and 14-round LBlock are 2^{-53} and 2^{-65} [56], respectively.
5. We obtain an 8-round related-key differential characteristic of DESL with probability $2^{-34.78}$ *in 10 minutes on a PC*. To the best of our knowledge, no

related-key differential characteristic covering more than 7 rounds of DESL has been published before.

6. We obtain a 7-round related-key characteristic for PRESENT-128 with probability 2^{-11} and 0 active S-box in its key schedule algorithm, based on which an improved related-key boomerang distinguisher for the 14-round PRESENT-128 and a key-recovery attack on the 17-round PRESENT-128 can be constructed by using exactly the same method presented in [47].

The method presented in this paper is generic, automatic, and applicable to other lightweight ciphers with bit-oriented operations. Due to the page limit, the concrete results concerning the related-key or single-key differential characteristics for LBlock, PRESENT-128, and DES(L) are put into an extended version of this paper available at <http://eprint.iacr.org/2013/676>.

Organization of the Paper. In Sect. 2, we introduce Mouha *et al.*'s framework and its extension for counting the number of active S-boxes of bit-oriented ciphers automatically with the MILP technique. In Sect. 3, we introduce the concept of valid cutting-off inequalities for tightening the feasible region of an MILP problem, and explore how to generate and select valid cutting-off inequalities. We present the methods for automatic security evaluation with respect to the (related-key) differential attack, and searching for (related-key) differential characteristics in Sect. 4 and Sect. 5. In Sect. 6 we conclude the paper and propose some research directions for bit-oriented ciphers and the application of the MILP technique in cryptography. The application of the methods presented in this paper to PRESENT, LBlock, and SIMON is given in Appendices.

2 Mouha *et al.*'s Framework and Its Extension

2.1 Mouha *et al.*'s Framework for Word-Oriented Block Ciphers

Assume a cipher is composed of the following three word-oriented operations, where ω is the word size:

- XOR, $\oplus : \mathbb{F}_2^\omega \times \mathbb{F}_2^\omega \rightarrow \mathbb{F}_2^\omega$;
- Linear transformation $L : \mathbb{F}_2^{m\omega} \rightarrow \mathbb{F}_2^{m\omega}$ with branch number \mathcal{B}_L ;
- S-box, $\mathcal{S} : \mathbb{F}_2^\omega \rightarrow \mathbb{F}_2^\omega$.

Mouha *et al.*'s framework uses 0-1 variables, which are subjected to certain constraints imposed by the above operations, to denote the word level differences propagating through the cipher (1 for nonzero difference and 0 otherwise).

Firstly, we should include the constraints imposed by the operations of the cipher.

Constraints Imposed by XOR Operations. Suppose $a \oplus b = c$, where $a, b, c \in \mathbb{F}_2^\omega$ are the input and output differences of the XOR operation, the following constraints will make sure that when a, b , and c are not all zero, then there are at least two of them are nonzero:

$$\begin{cases} a + b + c \geq 2d_\oplus \\ d_\oplus \geq a, d_\oplus \geq b, d_\oplus \geq c \end{cases} \quad (1)$$

where d_{\oplus} is a dummy variable taking values from $\{0, 1\}$. If each one of a , b , and c represents one bit, we should also add the inequality $a + b + c \leq 2$.

Constraints Imposed by Linear Transformation. Let x_{i_k} and y_{j_k} , $k \in \{0, 1, \dots, m-1\}$, be 0-1 variables denoting the word-level input and output differences of the linear transformation L respectively. Since for nonzero input differences, there are totally at least \mathcal{B}_L nonzero ω -bit words in the input and output differences, we include the following constraints:

$$\begin{cases} \sum_{k=0}^{m-1} (x_{i_k} + y_{j_k}) \geq \mathcal{B}_L d_L \\ d_L \geq x_{i_k}, d_L \geq y_{j_k}, \quad k \in \{0, \dots, m-1\} \end{cases} \quad (2)$$

where d_L is a dummy variable taking values in $\{0, 1\}$ and \mathcal{B}_L is the branch number of the linear transformation.

Then, we set up the **objective function** to be the sum of all variables representing the input words of the S-boxes.

2.2 Extension of Mouha *et al.*'s Framework for Bit-Oriented Ciphers

For bit-oriented ciphers, bit-level representations and additional constraints are needed [48]. For every input and output bit-level difference, a new 0-1 variable x_i is introduced such that $x_i = 1$ if and only if the difference at this bit is nonzero.

For every S-box in the schematic diagram, including the encryption process and the key schedule algorithm, we introduce a new 0-1 variable A_j such that $A_j = 1$ if the input word of the Sbox is nonzero and $A_j = 0$ otherwise.

At this point, it is natural to choose the objective function f , which will be minimized, as $\sum A_j$ for the goal of determining a lower bound of the number of active S-boxes.

For bit-oriented ciphers, we need to include two sets of constraints. The first one is the set of constraints imposed by XOR operations, and the other is due to the S-box operation. After changing the representations to bit-level, the set of constraints imposed by XOR operations for bit-oriented ciphers are the same as that presented in (1). The S-box operation is more tricky.

Constraints Describing the S-box Operation. Suppose $(x_{i_0}, \dots, x_{i_{\omega-1}})$ and $(y_{j_0}, \dots, y_{j_{\nu-1}})$ are the input and output bit-level differences of an $\omega \times \nu$ S-box marked by A_t . Firstly, to ensure that $A_t = 1$ holds if and only if $x_{i_0}, \dots, x_{i_{\omega-1}}$ are not all zero, we require that:

$$\begin{cases} A_t - x_{i_k} \geq 0, \quad k \in \{0, \dots, \omega-1\} \\ x_{i_0} + x_{i_1} + \dots + x_{i_{\omega-1}} - A_t \geq 0 \end{cases} \quad (3)$$

For bijective S-boxes, nonzero input difference must result in nonzero output difference and vice versa:

$$\begin{cases} \omega y_{j_0} + \omega y_{j_1} + \dots + \omega y_{j_{\nu-1}} - (x_{i_0} + x_{i_1} + \dots + x_{i_{\omega-1}}) \geq 0 \\ \nu x_{i_0} + \nu x_{i_1} + \dots + \nu x_{i_{\omega-1}} - (y_{j_0} + y_{j_1} + \dots + y_{j_{\nu-1}}) \geq 0 \end{cases} \quad (4)$$

Note that the above constraints should not be used for non-bijective S-box such as the S-box of DES(L) [37].

Finally, the Hamming weight of the $(\omega + \nu)$ -bit word $x_{i_0} \cdots x_{i_{\omega-1}} y_{j_0} \cdots y_{j_{\nu-1}}$ is lower bounded by the branch number \mathcal{B}_S of the S-box for nonzero input difference $x_{i_0} \cdots x_{i_{\omega-1}}$, where d_S is a dummy variable:

$$\begin{cases} \sum_{k=0}^{\omega-1} x_{i_k} + \sum_{k=0}^{\nu-1} y_{j_k} \geq \mathcal{B}_S d_S \\ d_S \geq x_{i_k}, d_S \geq y_{j_t}, \quad k \in \{0, \dots, \omega - 1\}, \quad t \in \{0, \dots, \nu - 1\} \end{cases} \quad (5)$$

where the branch number \mathcal{B}_S of an S-box \mathcal{S} , is defined as $\mathcal{B}_S = \min_{a \neq b} \{\text{wt}((a \oplus b) \| (\mathcal{S}(a) \oplus \mathcal{S}(b)) : a, b \in \mathbb{F}_2^\omega)\}$, and $\text{wt}(\cdot)$ is the standard Hamming weight of an $(\omega + \nu)$ -bit word. We point out that constraint (5) is redundant for an invertible S-box with branch number $\mathcal{B}_S = 2$, since in this particular case, all differential patterns not satisfying (5) violate (4).

0-1 Variables. The MILP model proposed above is indeed a Pure Integer Programming Problem since all variables appearing are 0-1 variables. However, in practice we only need to explicitly restrict a part of all variables to be 0-1, while all other variables can be allowed to be any real numbers, which leads to an MILP problem. Following this approach, the MILP solving process may be accelerated as suggested in [17].

3 Tighten the Feasible Region with Valid Cutting-off Inequalities

The feasible region of an MILP problem is defined as the set of all variable assignments satisfying all constraints in the MILP problem. The modelling process presented in the previous sections indicates that every differential path corresponds to a solution in the feasible region of the MILP problem. However, a feasible solution of the MILP model is not guaranteed to be a valid differential path, since our constraints are far from perfect to rule out all invalid differential patterns. For instance, assume x_i and y_i ($0 \leq i \leq 3$) are the bit-level input and output differences of the PRESENT-80 S-box. According to Sect. 2.2, x_i, y_i are subjected to the constraints of (3), (4) and (5). Obviously, $(x_0 \cdots, x_3, y_0, \cdots, y_3) = (1, 0, 0, 1, 1, 0, 1, 1)$ satisfies the above constraints, whereas $0x9 = 1001 \rightarrow 0xB = 1011$ is not a valid difference propagation pattern for the PRESENT S-box, which can be seen from the differential distribution table of the PRESENT S-box. Hence, we are actually trying to minimize the number of the active S-boxes over a larger region, and the optimum value obtained in this setting must be smaller than or equal to the actual minimum number of active S-boxes. Although the above fact will not invalidate the lower bound we obtained from our MILP model, this prevents the designers or analysts from obtaining tighter security bounds and valid (related-key) differential characteristics from the feasible region.

The situation would be even worse when modelling an invertible S-box with branch number $\mathcal{B}_S = 2$, which is the minimal value of the branch number for an invertible S-box. In the case of invertible S-box with $\mathcal{B}_S = 2$, the constraints of (3), (4) are enough, and (5) is redundant.

Therefore, we are motivated to look for linear inequalities which can cut off some part of the feasible region of the MILP model while leaving the region of valid differential characteristics intact. For the convenience of discussion, we give the following definition.

Definition 1. *A valid cutting-off inequality is a linear inequality which is satisfied by all possible valid differential patterns, but is violated by at least one feasible solution corresponding to an impossible differential pattern in the feasible region of the original MILP problem.*

3.1 Methods for Generating Valid Cutting-Off Inequalities

In this section, we present two methods for generating valid cutting-off inequalities by analyzing the differential behavior of the underlying S-box.

Modelling Conditional Differential Behaviour. In building integer programming models in practice, sometimes it is possible to model certain logical constraints as linear inequalities. For example, assume x is a continuous variable such that $0 \leq x \leq M$, where M is a fixed integer, and we know that δ is a 0-1 variable taking value 1 when $x > 0$, that is $x > 0 \Rightarrow \delta = 1$. It is easy to verify that the above logical condition can be achieved by imposing the constraint $x - M\delta \leq 0$.

In fact, there is a surprisingly large number of different types of logical conditions can be imposed in a similar way. We now give a theorem which will be used in the following.

Theorem 1. *If we assume that all variables are 0-1 variables, then the logical condition that $(x_0, \dots, x_{m-1}) = (\delta_0, \dots, \delta_{m-1}) \in \{0, 1\}^m \subseteq \mathbb{Z}^m$ implies $y = \delta \in \{0, 1\} \subseteq \mathbb{Z}$ can be described by the following linear inequality*

$$\sum_{i=0}^{m-1} (-1)^{\delta_i} x_i + (-1)^{\delta+1} y - \delta + \sum_{i=0}^{m-1} \delta_i \geq 0, \tag{6}$$

where δ_i, δ are fixed constants and \mathbb{Z} is the set of all integers.

Proof. We only prove the Theorem for the case $\delta = 0$. For $\delta = 1$, it can be proved in a similar way. We assume

$$(\delta_0, \dots, \delta_{m-1}) = (\delta_0, \dots, \delta_{s_1-1}; \delta_{s_1}, \dots, \delta_{m-1}) = (1, 1, \dots, 1; 0, 0, \dots, 0) = \Delta^*.$$

For other 0-1 patterns, it can be permuted into such a form and this will not affect our proof.

Firstly, $(\Delta^*, 0)$ is satisfied by (6), which can be verified directly.

Secondly, we prove that all vectors $(x_0, \dots, x_{m-1}, y) \in \{0, 1\}^{m+1}$ such that $(x_0, \dots, x_{m-1}) \neq \Delta^*$ are satisfied by (6). In such cases, we have

$$\sum_{i=0}^{m-1} (-1)^{\delta_i} x_i + (-1)^{\delta+1} y - \delta + \sum_{i=0}^{m-1} \delta_i = - \sum_{i=0}^{s_1-1} x_i + \sum_{i=s_1}^m x_i - y - 0 + s_1 \geq 0,$$

for $y = 0$ or $y = 1$.

Finally we prove that the vector $(x_0, \dots, x_{m-1}, y) = (\Delta^*, 1)$ is not satisfied by the linear inequality. In such case, we have

$$\sum_{i=0}^{m-1} (-1)^{\delta_i} x_i + (-1)^{\delta+1} y - \delta + \sum_{i=0}^{m-1} \delta_i = - \sum_{i=0}^{s_1-1} x_i + \sum_{i=s_1}^m x_i - 1 - 0 + s_1 < 0.$$

The proof is completed.

For example, the PRESENT S-box has the following conditional differential [26,32,38,33,18] properties, which are referred to as undisturbed bits in [50].

Fact 1. *The S-box of PRESENT-80 has the following properties:*

(i) $1001 \rightarrow ***0$: *If the input difference of the S-box is $0x9 = 1001$, then the least significant bit of the output difference must be 0;*

(ii) $0001 \rightarrow ***1$ and $1000 \rightarrow ***1$: *If the input difference of the S-box is $0x1 = 0001$ or $0x8 = 1000$, then the least significant bit of the output difference must be 1;*

(iii) $***1 \rightarrow 0001$ and $***1 \rightarrow 0100$: *If the output difference of the S-box is $0x1 = 0001$ or $0x4 = 0100$, then the least significant bit of the input difference must be 1; and*

(iv) $***0 \rightarrow 0101$: *If the output difference of the S-box is $0x5 = 0101$, then the least significant bit of the input difference must be 0.*

From Theorem 1, we have the following fact.

Fact 2. *Let 0-1 variables (x_0, x_1, x_2, x_3) and (y_0, y_1, y_2, y_3) represent the input and output bit-level differences of the S-box respectively, where x_3 and y_3 are the least significant bits. Then the logical conditions in Theorem 1 can be described by the following linear inequalities:*

$$-x_0 + x_1 + x_2 - x_3 - y_3 + 2 \geq 0 \tag{7}$$

$$\begin{cases} x_0 + x_1 + x_2 - x_3 + y_3 \geq 0 \\ -x_0 + x_1 + x_2 + x_3 + y_3 \geq 0 \end{cases} \tag{8}$$

$$\begin{cases} x_3 + y_0 + y_1 + y_2 - y_3 \geq 0 \\ x_3 + y_0 - y_1 + y_2 + y_3 \geq 0 \end{cases} \tag{9}$$

$$-x_3 + y_0 - y_1 + y_2 - y_3 + 2 \geq 0 \tag{10}$$

For example, the linear inequality (7) removes all differential patterns of the form $(x_0, \dots, x_3, y_0, \dots, y_3) = (1, 0, 0, 1, *, *, *, 1)$, where (x_0, \dots, x_3) and (y_0, \dots, y_3) are the input and output differences of the PRESENT S-box respectively. We call this group of constraints presented in (7), (8), (9), and (10) the constraints of conditional differential propagation (CDP constraints for short). The CDP constraints obtained from Fact 1 and the differential patterns removed by these CDP constraints are given in Table 1.

Table 1. Impossible differential patterns removed by the CDP constraints generated according to the differential properties of the PRESENT S-box. Here, a vector $(\lambda_0, \dots, \lambda_3, \gamma_0, \dots, \gamma_3, \theta)$ in the left column denotes a linear inequality $\lambda_0x_0 + \dots + \lambda_3x_3 + \gamma_0y_0 + \dots + \gamma_3y_3 + \theta \geq 0$.

Constraints obtained by logical condition modelling	Impossible differential patterns removed
$(-1, 1, 1, -1, 0, 0, 0, -1, 2)$	$(1, 0, 0, 1, 0, 0, 0, 0, 1), (1, 0, 0, 1, 0, 0, 0, 1, 1), (1, 0, 0, 1, 0, 1, 0, 0, 1), (1, 0, 0, 1, 0, 1, 1, 1, 1), (1, 0, 0, 1, 1, 0, 0, 0, 1), (1, 0, 0, 1, 1, 0, 0, 1, 1), (1, 0, 0, 1, 1, 1, 0, 0, 1), (1, 0, 0, 1, 1, 1, 1, 1, 1)$
$(1, 1, 1, -1, 0, 0, 0, 1, 0)$	$(0, 0, 0, 1, 0, 0, 0, 0, 0), (0, 0, 0, 1, 0, 0, 1, 0, 0), (0, 0, 0, 1, 0, 1, 0, 0, 0), (0, 0, 0, 1, 0, 1, 1, 0, 0), (0, 0, 0, 1, 1, 0, 0, 0, 0), (0, 0, 0, 1, 1, 0, 1, 0, 0), (0, 0, 0, 1, 1, 1, 0, 0, 0), (0, 0, 0, 1, 1, 1, 1, 0, 0)$
$(-1, 1, 1, 1, 0, 0, 0, 1, 0)$	$(1, 0, 0, 0, 0, 0, 0, 0, 0), (1, 0, 0, 0, 0, 0, 1, 0, 0), (1, 0, 0, 0, 0, 1, 0, 0, 0), (1, 0, 0, 0, 0, 1, 1, 0, 0), (1, 0, 0, 0, 1, 0, 0, 0, 0), (1, 0, 0, 0, 1, 0, 1, 0, 0), (1, 0, 0, 0, 1, 1, 0, 0, 0), (1, 0, 0, 0, 1, 1, 1, 0, 0)$
$(0, 0, 0, 1, 1, 1, 1, -1, 0)$	$(0, 0, 0, 0, 0, 0, 0, 0, 1), (0, 0, 1, 0, 0, 0, 0, 0, 1), (0, 1, 0, 0, 0, 0, 0, 0, 1), (0, 1, 1, 0, 0, 0, 0, 0, 1), (1, 0, 0, 0, 0, 0, 0, 0, 1), (1, 0, 1, 0, 0, 0, 0, 0, 1), (1, 1, 0, 0, 0, 0, 0, 0, 1), (1, 1, 1, 0, 0, 0, 0, 0, 1)$
$(0, 0, 0, 1, 1, -1, 1, 1, 0)$	$(0, 0, 0, 0, 0, 1, 0, 0, 0), (0, 0, 1, 0, 0, 1, 0, 0, 0), (0, 1, 0, 0, 0, 1, 0, 0, 0), (0, 1, 1, 0, 0, 1, 0, 0, 0), (1, 0, 0, 0, 0, 1, 0, 0, 0), (1, 0, 1, 0, 0, 1, 0, 0, 0), (1, 1, 0, 0, 0, 1, 0, 0, 0), (1, 1, 1, 0, 0, 1, 0, 0, 0)$
$(0, 0, 0, -1, 1, -1, 1, -1, 2)$	$(0, 0, 0, 1, 0, 1, 0, 1, 0), (0, 0, 1, 1, 0, 1, 0, 1, 0), (0, 1, 0, 1, 0, 1, 0, 1, 0), (0, 1, 1, 1, 0, 1, 0, 1, 0), (1, 0, 0, 1, 0, 1, 0, 1, 0), (1, 0, 1, 1, 0, 1, 0, 1, 0), (1, 1, 0, 1, 0, 1, 0, 1, 0), (1, 1, 1, 1, 0, 1, 0, 1, 0)$

However, there are cases where no such conditional differential property exists. For example, two out of the eight S-boxes of Serpent [6] exhibit no such property. Even when the S-box under consideration can be described with this logical condition modelling technique, the inequalities generated may be not enough to produce a satisfied result. In the following, a more general approach for generating valid cutting-off inequalities is proposed.

Convex Hull of All Possible Differentials for an S-box. The convex hull of a set Q of discrete points in \mathbb{R}^n is the smallest convex set that contains Q . A convex hull in \mathbb{R}^n can be described as the common solutions of a set of finitely many linear (in)equalities as follows:

$$\begin{cases} \lambda_{0,0}x_0 + \dots + \lambda_{0,n-1}x_{n-1} + \lambda_{0,n} \geq 0 \\ \dots \\ \gamma_{0,0}x_0 + \dots + \gamma_{0,n-1}x_{n-1} + \gamma_{0,n} = 0 \\ \dots \end{cases} \tag{11}$$

This is called the H-Representation of a convex hull. Computing the H-representation of the convex hull of a set of finitely many points is a fundamental algorithm in computation geometry with many applications.

If we treat a differential of an $\omega \times \nu$ S-box as a point in $\mathbb{R}^{\omega+\nu}$, then we can get a set of finitely many discrete points which includes all possible differential patterns of this S-box. For example, one possible differential pattern of PRESENT S-box is $0x9 = 1001 \rightarrow 0xE = 1110$ which is identified with $(1, 0, 0, 1, 1, 1, 1, 0)$. The set of all possible differential patterns for the S-boxes are essentially sets of finitely many discrete points in high dimensional space, hence we can compute their convex hulls by standard method in computational geometry.

We now define the convex hull of a specific $\omega \times \nu$ S-box to be the set of all linear (in)equalities in the H-Representation of the convex hull $\mathcal{V}_S \subseteq \mathbb{R}^{\omega+\nu}$ of all possible differential patterns of the S-box. The convex hull of a specific S-box can be obtained by using the `inequality_generator()` function in the `sage.geometry.polyhedron` class of the SAGE computer algebra system [49]. The convex hull of the PRESENT S-box contains 327 linear inequalities. Any one of these inequalities can be taken as a valid cutting-off inequality.

3.2 Selecting Valid Cutting-off Inequalities from the Convex Hull: A Greedy Approach

The number of (in)equalities in the H-Representation of a convex hull computed from a set of discrete points in n dimensional space is very large in general. For instance, the convex hull $\mathcal{V}_S \subseteq \mathbb{R}^8$ of a 4×4 S-box typically involves several hundreds of linear inequalities. Adding all of them to an MILP problem will make the MILP problem insolvable in practical time. Hence, it is necessary to select a small number, say n , of “best” inequalities from the convex hull. Here by “best” we mean that, among all possible selections of n inequalities, the selected ones maximize the number of removed impossible differentials. Obviously, this is a hard combinatorial optimization problem. Therefore, we design a greedy algorithm, listed in Algorithm 1, to approximate the optimum selection.

This algorithm builds up a set of valid cutting-off inequalities by selecting at each step an inequality from the convex hull which maximizes the number of removed impossible differential patterns from the current feasible region. For instance, We select 6 valid cutting-off inequalities from the convex hull of the PRESENT S-box using Algorithm 1. Compared with the 6 valid cutting-off inequalities obtained by Theorem 1 (see Table 1), they cut off 24 more impossible differential patterns, which leads to a relatively tighter feasible region.

Algorithm 1. Selecting n inequalities from the convex hull \mathcal{H} of an S-box

Input: \mathcal{H} : the set of all inequalities in the H-representation of the convex hull of an S-box; \mathcal{X} : the set of all impossible differential patterns of an S-box; n : a positive integer.

Output: \mathcal{O} : a set of n inequalities selected from \mathcal{H}

```

1  $l^* := \text{None}$ ;  $\mathcal{X}^* := \mathcal{X}$ ;  $\mathcal{H}^* := \mathcal{H}$ ;  $\mathcal{O} := \emptyset$ ;
2 for  $i \in \{0, \dots, n-1\}$  do
3    $l^* :=$  The inequality in  $\mathcal{H}^*$  which maximizes the number of removed
   impossible differential patterns from  $\mathcal{X}^*$  ;
4    $\mathcal{X}^* := \mathcal{X}^* - \{\text{removed impossible differential patterns by } l^*\}$ ;
5    $\mathcal{H}^* := \mathcal{H}^* - \{l^*\}$ ;  $\mathcal{O} := \mathcal{O} \cup \{l^*\}$ ;
6 end
7 return  $\mathcal{O}$ 

```

4 Automatic Security Evaluation

To obtain the security bound of a block cipher with respect to related-key differential attack, we can build an MILP model according to Sect. 2 with the constraints introduced in Sect. 3.1 and Sect. 3.2 included. Then we solve the MILP model using any MILP optimizer, and the optimized solution, say N , is the minimum number of the active S-boxes from which we can deduce that the probability of the best differential characteristic is upper bounded by ϵ^N , where ϵ is the maximum differential probability (MDP) a single S-box.

However, it is computationally infeasible to solve an MILP model generated by an r -round block cipher with large r . In such case, we can turn to the so called simple split approach. We split the r -round block cipher into two parts with consecutive r_1 rounds and r_2 rounds such that $r_1 + r_2 = r$. Then we apply our method to these two parts. Assuming that there are at least N_{r_1} and N_{r_2} active S-boxes in the first and second part respectively, we can deduce that the probability of the best differential characteristic for this r -round cipher is upper bounded by $\epsilon^{(N_{r_1} + N_{r_2})}$. If r_1 and r_2 are still too large, they can be divided into smaller parts further. Note that our method is applicable to both the single-key and related-key models.

4.1 Techniques for Getting Tighter Security Bounds

Technique 1. In the above analysis, we pessimistically (in the sense that we want to prove the security of a cipher) assume that all the active S-boxes take the MDP ϵ . However, this is unlikely to happen in practice, especially in the case that the number of active S-boxes is minimized. Therefore, we have the following strategy for obtaining tighter security bound for a t -round characteristic.

Firstly, compute the set \mathcal{E} of all the differential patterns of an S-box with probabilities greater than or equal to the S-box's MDP ϵ .

Secondly, compute the H-representation $H_{\mathcal{E}}$ of the convex hull of \mathcal{E} , and then use the inequalities selected from $H_{\mathcal{E}}$ by Algorithm 1 to generate a t -round

model according to Sect. 2 and Sect. 3. Note that the feasible region of this model is smaller than that of a t -round model generated in standard way, since the differential patterns allowed to take in this model is more restrictive. Hence, we hope to get a larger objective value than N_t , which is the result obtained by using the standard t -round model.

Finally, solve the model using a software optimizer. If the objective value is greater than N_t , we know that there is no differential characteristic with only N_t active S-boxes such that all these S-boxes take differential patterns with probability ϵ . And hence, we can conclude that there is at least one active S-box taking a differential pattern with probability less than ϵ in a t -round characteristic with only N_t active S-boxes.

Technique 2. Yet another technique for obtaining tighter security bound is inspired by Alex Biryukov *et al.* and Sareh Emami *et al.*'s (extended) split approach [11,22]. In Sun *et al.*'s work [48], the strategy for proving the security of an n -round iterative cipher against the related-key differential attacks is to use the simple split approach. By employing the MILP technique, compute the minimum number N_t of differentially active S-boxes for any consecutive t -round ($1 \leq t \leq n$) related-key differential characteristic. Then the lower bound of the number of active S-boxes for the full cipher (n -round) can be obtained by computing $\sum_{j \in I \subseteq \{1,2,\dots\}} N_{t_j}$, where $\sum_{j \in I} t_j = n$. Note that the computational cost is too high to compute N_n directly.

We point out that this simple “split strategy” can be improved to obtain tighter security bound by exploiting more information of a differential characteristic. The main idea is that the characteristic covering round 1 to round m and the characteristic covering round $m+1$ to round $2m$ should not be treated equal although they have the same number of rounds, since the starting difference of a characteristic of round $m+1$ to $2m$ is not as free as that of a characteristic of round 1 to round m . Therefore, we have the following strategy.

Firstly, split an r -round into two parts: round 1 to round r_1 , and round r_1+1 to round $r = r_1 + r_2$.

Secondly, construct an MILP model covering round 1 to round r . Change the objective function to be the sum of all S-boxes covering round r_1+1 to round r . Add some additional constraints on the number of active S-boxes covering round 1 to round r_1 (One way to obtain such constraints is to solve the model covering round 1 to round r_1).

Finally, solve the model using any software optimizer, and the result is the lower bound of the number of active S-boxes of round r_1+1 to round r (r_2 rounds in total) for any characteristic covering round 1 to round r .

We have applied the methods presented in this section to PRESENT-80 and LBlock, and the results are given in Appendix A.

5 A Heuristic Method for Finding (Related-key) Differential Characteristics Automatically

To find a (related-key) differential characteristic with relatively high probability covering r rounds of a cipher is the most important step in (related-key) differential cryptanalysis. Most of the tools for searching differential characteristics are essentially based on Matsui's algorithm [42]. In this section, we propose an MILP based heuristic method for finding (related-key) differential characteristics. Compared to other methods, our method is easier to implement, and more flexible.

Thanks to the valid cutting-off inequalities which can describe the property of an S-box according to its differential distribution table, our method can output a good (related-key) differential characteristic directly by employing the MILP technique. The procedure of our method is outlined as follows.

Step 1. For every S-box \mathcal{S} , select n inequalities from the convex hull of the set of all possible differential patterns of \mathcal{S} using Algorithm 1, and generate an r -round MILP model in which we require that *all variables involved are 0-1*.

Step 2. Extract a feasible solution of the MILP model by using the Gurobi [45] optimizer.

Step 3. Check whether the feasible solution is a valid (related-key) differential characteristic. If it is a valid characteristic, the procedure terminates. Otherwise, go to step 1, increase the number of selected inequalities from the convex hulls, and repeat the whole process.

We have developed a software by employing the python interface provided by the Gurobi optimizer, which automates the whole process of the above method.

To demonstrate the practicability of our method, we have applied the methods presented in this section to SIMON and the results are given in Appendix B.

On the Quality of the Characteristics. The characteristics found by this method are not guaranteed to be the best. However, if you would like to wait until the optimizer outputs optimum solution, the characteristic found by this method is guaranteed to have the minimum number of active S-boxes. Experimental results show that we get reasonably good solutions.

On the Flexibility of the Searching Algorithm. By adding a small number of additional constraints, our method can be used to search characteristics with specific properties. For example, by setting some given variables marking the activity of some S-boxes to 1, we can search for characteristics with active S-boxes of predefined positions, which may be used in leaked-state forgery attacks [55]; by requiring the output and input variables to be the same, we can search for iterative characteristics; by setting all the variables marking the activity of all the S-boxes in the key schedule algorithm to be 0, we can search for characteristics with 0 active S-boxes in its key schedule algorithm, which may be preferred in the related-key differential attack.

6 Conclusion and Directions for Future Work

In this paper, we bring new constraints into the MILP model to describe the differential properties of a specific S-box, and obtain a more accurate MILP model for the differential behavior of a block cipher. Based on these constraints, we propose an automatic method for evaluating the security of bit-oriented block ciphers with respect to (related-key) differential attack. We also present a new tool for finding (related-key) characteristics automatically.

At this point, several open problems emerge. Firstly, we observe that the MILP instances derived from such cryptographic problems are very hard to solve compared with general MILP problems with the same scale with respect to the numbers of variables and constraints. Hence, it is interesting to develop specific methods to accelerate the solving process of such problems and therefore increase the number of rounds of the cipher under consideration that can be dealt with. Secondly, the method presented in this paper is very general. Is it possible to develop a compiler which can convert a standard description, say a description using hardware description language, of a cipher into an MILP instance to automate the entire security evaluation cycle with respect to (related-key) differential attack?

Finally, the methodology presented in this paper has some limitations which we would like to make clear, and trying to overcome these limitations is a topic deserving further investigation. Firstly, this methodology is only suitable to evaluate the security of constructions with S-boxes, XOR operations and bit permutations, and can not be applied to block cipher like SPECK [5], which involve modulo addition and no S-boxes at all. For tools which can be applied to ARX constructions, we refer the reader to [12,39,40,41,43]. Secondly, in this paper we do not consider the *differential effect* and we assume that the expected differential probability (EDP) π of a characteristic over all keys is (almost) the same as the fixed-key differential probability (DP) π_K for almost all keys (the common hypothesis of *stochastic equivalence* [35]), and that if the lower bound of the EDP for any characteristic of a block cipher is less than 2^{-s} , where s is bigger than the block size or key size, then the block cipher is secure against the (related-key) differential attack. For more in-depth discussion of the essential gap between EDP π and DP π_K , we refer the reader to [13] for more information.

Acknowledgements. The authors would like to thank the anonymous reviewers for their helpful comments and suggestions. The work of this paper is supported by the National Key Basic Research Program of China (2013CB834203, 2014CB340603), the National Natural Science Foundation of China (Grants 61402469, 61472417, 61472415 and 61272477), the Strategic Priority Research Program of Chinese Academy of Sciences under Grant XDA06010702, and the State Key Laboratory of Information Security, Chinese Academy of Sciences.

References

1. Albrecht, M., Cid, C.: Cold boot key recovery by solving polynomial systems with noise. In: Lopez, J., Tsudik, G. (eds.) ACNS 2011. LNCS, vol. 6715, pp. 57–72. Springer, Heidelberg (2011)
2. Biryukov, A., Roy, A., Velichkov, V.: Differential analysis of block ciphers SIMON and SPECK. In: Fast Software Encryption, FSE 2014 (2014)
3. Alkhzaimi, H.A., Lauridsen, M.M.: Cryptanalysis of the SIMON family of block ciphers. Cryptology ePrint Archive, Report 2013/543 (2013), <http://eprint.iacr.org/2013/543>
4. Aoki, K., Ichikawa, T., Kanda, M., Matsui, M., Moriai, S., Nakajima, J., Tokita, T.: *Camellia*: A 128-bit block cipher suitable for multiple platforms - design and analysis. In: Stinson, D.R., Tavares, S. (eds.) SAC 2000. LNCS, vol. 2012, pp. 39–56. Springer, Heidelberg (2001)
5. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK families of lightweight block ciphers. Cryptology ePrint Archive, Report 2013/404 (2013), <http://eprint.iacr.org/2013/404>
6. Biham, E., Anderson, R., Knudsen, L.: Serpent: A new block cipher proposal. In: Vaudenay, S. (ed.) FSE 1998. LNCS, vol. 1372, pp. 222–238. Springer, Heidelberg (1998)
7. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. *Journal of Cryptology* 4(1), 3–72 (1991)
8. Bilgin, B., Bogdanov, A., Knežević, M., Mendel, F., Wang, Q.: FIDES: Lightweight authenticated cipher with side-channel resistance for constrained hardware. In: Bertoni, G., Coron, J.-S. (eds.) CHES 2013. LNCS, vol. 8086, pp. 142–158. Springer, Heidelberg (2013)
9. Biryukov, A.: Impossible differential attack. In: Encyclopedia of Cryptography and Security, pp. 597–597. Springer (2011)
10. Biryukov, A., Nikolić, I.: Automatic search for related-key differential characteristics in byte-oriented block ciphers: Application to AES, Camellia, Khazad and others. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 322–344. Springer, Heidelberg (2010)
11. Biryukov, A., Nikolić, I.: Search for related-key differential characteristics in DES-like ciphers. In: Joux, A. (ed.) FSE 2011. LNCS, vol. 6733, pp. 18–34. Springer, Heidelberg (2011)
12. Biryukov, A., Velichkov, V.: Automatic search for differential trails in ARX ciphers. In: Benaloh, J. (ed.) CT-RSA 2014. LNCS, vol. 8366, pp. 227–250. Springer, Heidelberg (2014)
13. Blondeau, C., Bogdanov, A., Leander, G.: Bounds in shallows and in miseries. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 204–221. Springer, Heidelberg (2013)
14. Bogdanov, A.: On unbalanced feistel networks with contracting MDS diffusion. *Designs, Codes and Cryptography* 59(1-3), 35–58 (2011)
15. Bogdanov, A.A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M., Seurin, Y., Vikkelsoe, C.: PRESENT: An ultra-lightweight block cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450–466. Springer, Heidelberg (2007)
16. Borghoff, J., Canteaut, A., Güneysu, T., Kavun, E.B., Knezevic, M., Knudsen, L.R., Leander, G., Nikov, V., Paar, C., Rechberger, C., Rombouts, P., Thomsen, S.S., Yalçın, T.: PRINCE – A low-latency block cipher for pervasive computing

- applications. In: Wang, X., Sako, K. (eds.) ASIACRYPT 2012. LNCS, vol. 7658, pp. 208–225. Springer, Heidelberg (2012)
17. Borghoff, J., Knudsen, L.R., Stolpe, M.: Bivium as a mixed-integer linear programming problem. In: Parker, M.G. (ed.) Cryptography and Coding 2009. LNCS, vol. 5921, pp. 133–152. Springer, Heidelberg (2009)
 18. Bulygin, S., Walter, M.: Study of the invariant coset attack on PRINTcipher: more weak keys with practical key recovery. Tech. rep., Cryptology ePrint Archive, Report 2012/85 (2012), <http://eprint.iacr.org/2012/085.pdf>
 19. Daemen, J., Rijmen, V., Proposal, A.: Rijndael. In: Proceedings from the First Advanced Encryption Standard Candidate Conference, National Institute of Standards and Technology (NIST) (1998)
 20. Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mendel, F., Mennink, B., Mouha, N., Wang, Q., Yasuda, K.: PRIMATES v1. CAESAR submission (2014), <http://competitions.cr.yj.to/round1/primatesv1.pdf>
 21. Kavun, E.B., Lauridsen, M.M., Leander, G., Rechberger, C., Schwabe, P., Yalcin, T.: Prøst v1. CAESAR submission (2014), <http://competitions.cr.yj.to/round1/proestv1.pdf>
 22. Emami, S., Ling, S., Nikolic, I., Pieprzyk, J., Wang, H.: The resistance of PRESENT-80 against related-key differential attacks. Cryptology ePrint Archive, Report 2013/522 (2013), <http://eprint.iacr.org/>
 23. Abed, F., List, E., Wenzel, J., Lucks, S.: Differential cryptanalysis of round-reduced SIMON and SPECK. In: Fast Software Encryption, FSE 2014 (2014)
 24. Abed, F., List, E., Lucks, S., Wenzel, J.: Differential and linear cryptanalysis of reduced-round SIMON. Cryptology ePrint Archive, Report 2013/526 (2013), <http://eprint.iacr.org/526/>
 25. Fouque, P.A., Jean, J., Peyrin, T.: Structural evaluation of AES and chosen-key distinguisher of 9-round AES-128. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 183–203. Springer, Heidelberg (2013)
 26. Fuhr, T.: Finding second preimages of short messages for Hamsi-256. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 20–37. Springer, Heidelberg (2010)
 27. Nikolić, I.: Tweaking AES. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) SAC 2010. LNCS, vol. 6544, pp. 198–210. Springer, Heidelberg (2011)
 28. Alizadeh, J., Bagheri, N., Gauravaram, P., Kumar, A., Sanadhya, S.K.: Linear cryptanalysis of round reduced SIMON. Cryptology ePrint Archive, Report 2013/663 (2013), <http://eprint.iacr.org/2013/663>
 29. Jean, J., Nikolić, I., Peyrin, T.: Deoxys v1. CAESAR submission (2014), <http://competitions.cr.yj.to/round1/deoxysv1.pdf>
 30. Jean, J., Nikolić, I., Peyrin, T.: Joltik v1. CAESAR submission (2014), <http://competitions.cr.yj.to/round1/joltikv1.pdf>
 31. Jean, J., Nikolić, I., Peyrin, T.: Kiasu v1. CAESAR submission (2014), <http://competitions.cr.yj.to/round1/kiasuv1.pdf>
 32. Knellwolf, S., Meier, W., Naya-Plasencia, M.: Conditional differential cryptanalysis of NLFSR-based cryptosystems. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 130–145. Springer, Heidelberg (2010)
 33. Knellwolf, S., Meier, W., Naya-Plasencia, M.: Conditional differential cryptanalysis of trivium and KATAN. In: Miri, A., Vaudenay, S. (eds.) SAC 2011. LNCS, vol. 7118, pp. 200–212. Springer, Heidelberg (2012)
 34. Knudsen, L.R.: Truncated and higher order differentials. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 196–211. Springer, Heidelberg (1995)

35. Lai, X., Massey, J.L.: Markov ciphers and differential cryptanalysis. In: Davies, D.W. (ed.) EUROCRYPT 1991. LNCS, vol. 547, pp. 17–38. Springer, Heidelberg (1991)
36. Lamberger, M., Nad, T., Rijmen, V.: Numerical solvers and cryptanalysis. *Journal of Mathematical Cryptology* 3(3), 249–263 (2009)
37. Leander, G., Paar, C., Poschmann, A., Schramm, K.: New lightweight DES variants. In: Biryukov, A. (ed.) FSE 2007. LNCS, vol. 4593, pp. 196–210. Springer, Heidelberg (2007)
38. Lehmann, M., Meier, W.: Conditional differential cryptanalysis of grain-128a. In: Pieprzyk, J., Sadeghi, A.-R., Manulis, M. (eds.) CANS 2012. LNCS, vol. 7712, pp. 1–11. Springer, Heidelberg (2012)
39. Leurent, G.: Construction of differential characteristics in ARX designs application to skein. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 241–258. Springer, Heidelberg (2013)
40. Lipmaa, H., Moriai, S.: Efficient algorithms for computing differential properties of addition. In: Matsui, M. (ed.) FSE 2001. LNCS, vol. 2355, pp. 336–350. Springer, Heidelberg (2002)
41. Lipmaa, H., Wallén, J., Dumas, P.: On the additive differential probability of exclusive-or. In: Roy, B., Meier, W. (eds.) FSE 2004. LNCS, vol. 3017, pp. 317–331. Springer, Heidelberg (2004)
42. Matsui, M.: On correlation between the order of S-boxes and the strength of DES. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 366–375. Springer, Heidelberg (1995)
43. Mouha, N., Preneel, B.: Towards finding optimal differential characteristics for ARX: Application to Salsa20. *Cryptology ePrint Archive, Report 2013/328* (2013), <http://eprint.iacr.org/2013/328>
44. Mouha, N., Wang, Q., Gu, D., Preneel, B.: Differential and linear cryptanalysis using mixed-integer linear programming. In: Wu, C.-K., Yung, M., Lin, D. (eds.) *Inscrypt 2011*. LNCS, vol. 7537, pp. 57–76. Springer, Heidelberg (2012)
45. Optimization, Gurobi: Gurobi optimizer reference manual (2013), <http://www.gurobi.com>
46. Oren, Y., Kirschbaum, M., Popp, T., Wool, A.: Algebraic side-channel analysis in the presence of errors. In: Mangard, S., Standaert, F.-X. (eds.) CHES 2010. LNCS, vol. 6225, pp. 428–442. Springer, Heidelberg (2010)
47. Özen, O., Varici, K., Tezcan, C., Kocair, Ç.: Lightweight block ciphers revisited: Cryptanalysis of reduced round PRESENT and HIGHT. In: Boyd, C., González Nieto, J. (eds.) ACISP 2009. LNCS, vol. 5594, pp. 90–107. Springer, Heidelberg (2009)
48. Sun, S., Hu, L., Song, L., Xie, Y., Wang, P.: Automatic security evaluation of block ciphers with s-bp structures against related-key differential attacks. In: *International Conference on Information Security and Cryptology – Inscrypt 2013* (2013)
49. Stein, W., et al.: Sage: Open source mathematical software (2008)
50. Tezcan, C.: Improbable differential attacks on PRESENT using undisturbed bits. *Journal of Computational and Applied Mathematics* 259, 503–511 (2014)
51. Wagner, D.: The boomerang attack. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, pp. 156–170. Springer, Heidelberg (1999)
52. Walter, M., Bulygin, S., Buchmann, J.: Optimizing guessing strategies for algebraic cryptanalysis with applications to EPCBC. In: Kutyłowski, M., Yung, M. (eds.) *Inscrypt 2012*. LNCS, vol. 7763, pp. 175–197. Springer, Heidelberg (2013)

53. Winnen, L.: Sage S-box MILP toolkit,
<http://www.ecrypt.eu.org/tools/sage-s-box-milp-toolkit>
54. Wu, S., Wang, M.: Security evaluation against differential cryptanalysis for block cipher structures. Tech. rep., Cryptology ePrint Archive, Report 2011/551 (2011),
<http://eprint.iacr.org/2011/551.pdf>
55. Wu, S., Wu, H., Huang, T., Wang, M., Wu, W.: Leaked-state-forgery attack against the authenticated encryption algorithm ALE. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 377–404. Springer, Heidelberg (2013)
56. Wu, W., Zhang, L.: LBlock: A lightweight block cipher. In: Lopez, J., Tsudik, G. (eds.) ACNS 2011. LNCS, vol. 6715, pp. 327–344. Springer, Heidelberg (2011)
57. Yap, H., Khoo, K., Poschmann, A., Henricksen, M.: EPCBC - A block cipher suitable for electronic product code encryption. In: Lin, D., Tsudik, G., Wang, X. (eds.) CANS 2011. LNCS, vol. 7092, pp. 76–97. Springer, Heidelberg (2011)
58. Sasaki, Y., Todo, Y., Aoki, K., Naito, Y., Sugawara, T., Murakami, Y., Matsui, M., Hirose, S.: Minalpher v1. CAESAR submission (2014),
<http://competitions.cr.yp.to/round1/minalpherv1.pdf>

A On the Security of PRESENT-80, and LBlock with Respect to the Related-Key Differential Attack

A.1 Results on PRESENT-80

We apply the logical condition modelling method presented in Sect. 3.1 to the block cipher PRESENT-80 to determine its security bound with respect to the related-key differential attack. In each of these MILP models, we include one more constraint to ensure that the difference of the initial key register is nonzero, since the case where the difference of the initial key register is zero can be analyzed in the single-key model. Then we employ the Gurobi 5.5 optimizer [45] to solve the MILP instances.

By default the computations are performed on a PC using 4 threads with Intel(R) Core(TM) Quad CPU (2.83GHz, 3.25GB RAM, Windows XP), and a star “*” is appended on a timing data to mark that the corresponding computation is taken on a workstation equipped with two Intel(R) Xeon(R) E5620 CPU(2.4GHz, 8GB RAM, 8 cores).

We compute the number of active S-boxes for PRESENT-80 in the related-key model up to 14 rounds, and the results and a comparison with previous results without using CDP constraints are summarized in Table 2. For example, according to the 6th row of Table 2, the Gurobi optimizer finds that the minimum number of active S-boxes for 6-round PRESENT-80 is at least 5 in no more than 16 seconds by solving the MILP model with CDP constraints

These results clearly demonstrate that the MILP models with CDP constraints lead to tighter security bounds. In particular, we have proved that there are at least 16 active S-boxes in the best related-key differential characteristic for any consecutive 12-rounds of PRESENT-80. Therefore, the probability of the best related-key differential characteristic of 24-round PRESENT-80 is $(2^{-2})^{16} \times (2^{-2})^{16} = 2^{-64}$, leading to the result that the 24-round PRESENT-80 is resistant to basic related-key differential attack based on related-key differential characteristic (rather than differential).

Table 2. Results obtained from MILP models for PRESENT-80

Rounds	With CDP Constraints		Without CDP Constraints	
	# Active S-boxes	# Time(in seconds)	# Active S-boxes	# Time(in seconds)
1	0	1	0	1
2	0	1	0	1
3	1	1	1	1
4	2	1	2	1
5	3	5	3	3
6	5	16	4	10
7	7	107	6	26
8	9	254	8	111
9	10	522	9	171
10	13	4158	12	1540
11	15	18124	13	8136
12	16	50017	15	18102
13	18	137160*	17	49537*
14	20	1316808*	18	685372*
15	–	> 20days	–	> 20days

A.2 Results on LBlock

Up to now, there is no concrete result concerning the security of full-round LBlock [56] against differential attack in the related-key model due to a lack of proper tools for bit-oriented designs.

Since the encryption process of LBlock is nibble-oriented, the security of LBlock against single-key differential attack can be evaluated by those word-oriented techniques. However, the “ $\lll 29$ ” operations in the key schedule algorithm of LBlock destroy its overall nibble-oriented structure. In this subsection, we apply the method proposed in this paper to LBlock, and some results concerning its security against related-key differential attacks are obtained. Note that the type of constraints given in (5) are removed in our MILP models for LBlock according to the explanations presented in previous sections.

From Table 3, we can deduce that the probability of the best differential characteristic for full LBlock (totally $32 = 11 + 11 + 10$ rounds) is upper bounded by $(2^{-2})^{10} \times (2^{-2})^{10} \times (2^{-2})^8 = 2^{-56}$, where 2^{-2} is the MDP for a single S-box of LBlock.

In fact, here we have an implicit trade-off between the number of constraints we use and the number of rounds we analyze. For example, we can use less constraints for every S-box and try to analyze more rounds, or we can use more constraints and focus on less rounds (but stronger bounds). However, it is not a simple task to find the best trade-off due to our limited computational power. We do try to analyze more rounds by using only one inequality selected from the convex hull for every S-box. The largest number of rounds we are able to analyze is 13, and we have prove that *there are at least 13 active S-boxes in any related-key characteristic for 13-round LBlock* on a PC in roughly 49 days.

Table 3. Results for related-key differential analysis on LBlock (The #Variables column records the sum of the number of the 0-1 variables and continuous variables in the MILP model).

Rounds	#Variables	#Constraints	#Active S-boxes	Time (in seconds)
1	218+104 = 322	660	0	1
2	292+208 = 500	1319	0	1
3	366+312 = 678	1978	0	1
4	440+416 = 856	2637	0	1
5	514+520 = 1034	3296	1	2
6	588+624 = 1212	3955	2	12
7	662+728 = 1390	4614	3	38
8	736+832 = 1568	5273	5	128
9	810+936 = 1746	5932	6	386
10	884+1040 = 1924	6591	8	19932
11	958+1144 = 2102	7250	10	43793

Then, we try to improve the above result with the two techniques presented in Sect. 4.1. By using the first technique, we can show that *there are at least 13 active S-boxes in a 13-round related-key differential characteristic of LBlock, and there is at least one active S-box taking a differential pattern with probability 2^{-3} in any 13-round related-key differential characteristic of LBlock with only 13 active S-boxes. Therefore, the probability of a 13-round related-key differential characteristic of LBlock is upper bounded by $(2^{-2})^{12} \times (2^{-3}) = 2^{-27}$.*

We now turn to the second technique presented in Sect. 4.1. By adding the constraint that the number of active S-boxes of any characteristic covering round 22 to round 26 (5 rounds in total) has at least 1 active S-box (see Table 3), and at most 12 active S-boxes to a 11-round (round 22 to round 32) MILP model (If this is not the case, it will enable us to get better bounds than the result presented here), we can show that there are at least 3 active S-boxes in a characteristic covering round 27 to round 32. Combined with Fact 3, we have that the probability of the best related-key differential characteristic for full LBlock is upper bounded by $2^{-27} \times 2^{-27} \times (2^{-2})^3 = 2^{-60}$.

B Search for Related-Key Characteristics of SIMON48

SIMON [5] is a family of lightweight block ciphers designed by the U.S National Security Agency (NSA). For a detailed description of SIMON and existing attacks, we refer the reader to [2,3,23,24,28].

By treating the AND ($\mathbb{F}_2 \times \mathbb{F}_2 \rightarrow \mathbb{F}_2$) operation as a 2×1 S-box, we apply our method to SIMON in the single-key model. For SIMON48 we obtain a 15-round differential characteristic with probability 2^{-46} (see Table 4), which is the best 15-round differential characteristic for 15-round SIMON48 published so far. If we fix the input and output differences to be the differences suggested by the characteristic we found, we can compute the probability of this *differential* by

searching all characteristics with probability greater than 2^{-54} in this differential, and the result is $2^{-41.96}$ which is also the best result published so far.

We would like to emphasize that in our MILP models we treat the input bits of the AND operation as *independent input bits*, and the dependencies of the input bits to the AND operation are not considered. Therefore, the characteristic obtained by our method is not guaranteed to be valid. Hence, every time after the Gurobi optimizer outputs a good solution (characteristic), we check its validity and compute its probability by the method presented in [2].

Table 4. Single-key differential characteristic of 15-round SIMON48

Rounds	Left	Right
0	000000001000000000000000	000000100010001000000000
1	000000000100010000000000	000000001000000000000000
2	000000000000100000000000	000000000100010000000000
3	000000000000010000000000	000000000001000000000000
4	000000000000000000000000	000000000000010000000000
5	000000000000001000000000	000000000000000000000000
6	000000100000100000000000	000000000000001000000000
7	000000000010001000000010	000000100000100000000000
8	001000001000001000001000	000000000010001000000010
9	000000000010001000000010	001000001000001000001000
10	000000100000100000000000	000000000010001000000010
11	000000000000001000000000	000000100000100000000000
12	000000000000000000000000	000000000000010000000000
13	000000000000001000000000	000000000000000000000000
14	000000000000100000000000	000000000000001000000000
15	000000000010001000000000	000000000000100000000000